

EXACT SUBSAMPLING MCMC

MATIAS QUIROZ, MINH-NGOC TRAN, MATTIAS VILLANI AND ROBERT KOHN

ABSTRACT. Speeding up Markov Chain Monte Carlo (MCMC) for data sets with many observations by data subsampling has recently received considerable attention in the literature. Most of the proposed methods are approximate, and the only exact solution has been documented to be highly inefficient. We propose a simulation consistent subsampling method for estimating expectations of any function of the parameters using a combination of MCMC with data subsampling and the importance sampling correction for occasionally negative likelihood estimates in Lyne et al. (2015). Our algorithm is based on first obtaining an unbiased but not necessarily positive estimate of the likelihood. The estimator uses a soft lower bound such that the likelihood estimate is positive with a high probability, and computationally cheap control variables to lower variability. Second, we carry out a correlated pseudo marginal MCMC on the absolute value of the likelihood estimate. Third, the sign of the likelihood is corrected using an importance sampling step that has low variance by construction. We illustrate the usefulness of the method with two examples.

KEYWORDS: Bayesian inference, Metropolis-Hastings, Pseudo-marginal MCMC, Exact inference, Estimated likelihood, Control variates, Data subsampling.

1. INTRODUCTION

Standard Markov Chain Monte Carlo (MCMC) algorithms require evaluating the likelihood function for the full data set and are therefore prohibitively expensive for so-called tall data sets with many observations. One recent strand of literature attempts to speed up MCMC algorithms by using random subsets of the data, see Korattikara et al. (2014); Bardenet et al. (2014, 2015); Maclaurin and Adams (2014); Liu et al. (2015); Quiroz et al. (2016). Section 2 briefly reviews these approaches and highlights possible pitfalls.

Quiroz and Kohn: *School of Economics, UNSW Business School, University of New South Wales*. Tran: *Discipline of Business Analytics, University of Sydney*. Villani: *Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University*.

In an excellent review of subsampling approaches, Bardenet et al. (2015) propose a positive unbiased estimator of the likelihood in a pseudo-marginal framework to accelerate the Metropolis-Hastings (MH) algorithm (Metropolis et al., 1953; Hastings, 1970). The unbiased likelihood estimator is obtained from a sequence of unbiased log-likelihood estimates, which are subsequently used to debias the estimator in the ordinary scale following Rhee and Glynn (2015). Moreover, to ensure positiveness a lower bound of the log-likelihood estimate is assumed (Jacob and Thiery, 2015). Bardenet et al. (2015) quickly dismiss this approach as the estimator’s large variability causes the pseudo-marginal chain to get stuck for long spells. Our article modifies their approach in three crucial ways to overcome its inherent inefficiency. First, we use the control variates in Quiroz et al. (2016) to reduce the variance of the likelihood estimator by several orders of magnitude. Second, we explore recent ideas in the Pseudo-Marginal Metropolis-Hastings (PMMH) literature and correlate the subsamples at the current and proposed draw when estimating the MH ratio of the PMMH algorithm. This implies that the method can tolerate a much larger variance of the likelihood estimator without getting stuck in the MCMC sampling. Third, we argue that the requirement of an exact lower bound is by construction inefficient, as a too large negative value is likely to adversely affect the variance of the estimator. We therefore introduce what we call a soft lower bound, which is a lower bound with a probability \tilde{p} controlled by the user. Setting this probability close to one results in a high probability of the likelihood estimator being positive. As in Lyne et al. (2015) the PMMH scheme instead targets an absolute measure, and the draws are corrected with an importance sampling step to estimate posterior expectations of arbitrary functions exactly. They show that a small number of negative signs of the estimate gives a small variance for the importance sampler. An attractive feature of our approach is that the probability for a positive estimator is a function of the lower bound: the choice of a high \tilde{p} results in few negative signs. However, making the lower bound too extreme might not be desirable for the variance of the estimator as discussed above, hence affecting the mixing of the Markov chain. Finally, in addition to the estimator in Rhee and Glynn (2015), we also propose to use the Poisson estimator (Wagner 1988; 1989; Beskos

et al., 2006; Papaspiliopoulos, 2009; Fearnhead et al., 2010) with a soft lower bound. The Poisson estimator is more tractable and has a smaller variance in general.

We demonstrate that our modifications are successful and indeed give an efficient sampler that does not get stuck and generates many more efficient draws for a given computational budget compared to MH on the full sample; this is especially true for the Poisson estimator. Our exact subsampling approach increases in the variance compared to the approximate approach in Quiroz et al. (2016) and therefore will typically require a larger subsample size for estimation. We believe that our method is a useful complement to the approximate subsampling MCMC framework proposed in Quiroz et al. (2016) if exact inference is of importance. On the other hand, if computational cost is the primary concern, then the approximate subsampling MCMC should be the main choice. In particular, we demonstrate that our exact subsampling MCMC approach is dramatically more efficient than the only exact algorithm proposed so far. This article is organized as follows. Section 2 discusses the main subsampling approaches proposed in the recent literature. Section 3 modifies the estimator initially proposed in Bardenet et al. (2015) and outlines the Poisson estimator. The same section also discusses the idea of a soft lower bound that we apply to both estimators. Section 4 outlines our proposed sampling algorithm. Section 5 demonstrates the methodology on two AR processes. Section 6 concludes and outlines future research.

2. PREVIOUS RESEARCH

Previous research in scalable MCMC by data subsampling is either approximate (Korattikara et al., 2014; Bardenet et al., 2014, 2015; Quiroz et al., 2016) or exact (Maclaurin and Adams, 2014; Liu et al., 2015). We also note that exact subsampling approaches within a delayed acceptance MCMC (Christen and Fox, 2005) method have been proposed (Banterle et al., 2014; Payne and Mallick, 2015; Quiroz, 2016), but since the full data set must be evaluated for any accepted sample, these methods are not fully subsampling approaches.

The algorithms in Korattikara et al. (2014); Bardenet et al. (2014, 2015) all replace the computationally costly MH ratio with a hypothesis test based on a fraction of the data,

thereby significantly speeding up computations. Bardenet et al. (2015) evaluate these methods and conclude that their method with concentration inequalities and control variates clearly outperforms the algorithms in Korattikara et al. (2014) and Bardenet et al. (2014). A drawback of their method is that it relies on a bound for the difference between the log-likelihood contributions at the proposed and current sample, and that of the control variates. Their proposed Taylor-Lagrange bound can result in a too rough upper bound, which then has to be compensated by a very large subsample before the accept (or reject) decision can be taken (Quiroz et al., 2016).

Quiroz et al. (2016) estimate the MH ratio based on a random subsample and rely on a pseudo-marginal approach to sample from the posterior. They derive an efficient unbiased log-likelihood estimator which uses control variates to homogenize the sampling population. The corresponding likelihood estimator is biased and a bias-correction is proposed. Quiroz et al. (2016) prove that the posterior targeted by their scheme is within $O(m^{-2})$ of the true posterior for a fixed number of observations n , where m is the subsample size. Moreover, they extend the correlated pseudo-marginal method in Deligiannidis et al. (2016); Dahlin et al. (2015) to a subsampling setting. In principle, this allows for a less precise estimator of the log-likelihood based on a much smaller subsample. The bias-correction implies that the approximation error is a function of the variance of the log-likelihood estimator, and targeting a large variance may therefore degrade the posterior approximation. However, in their examples the approximation error is small even when the variance of the log-likelihood estimator is large.

The only proposed exact method is Firefly Monte Carlo in Maclaurin and Adams (2014) (see also Liu et al., 2015). The method introduces an auxiliary variable for each observation which determines if it should be included in the evaluation of the posterior. A lower bound for each likelihood term caters for the observations that are not included in the evaluation of the posterior. The authors suggest using Gibbs sampling with the original parameters and the auxiliary variables in two different blocks. The method has been documented to be very inefficient, see e.g. Bardenet et al. (2015) and Quiroz et al. (2016), because of the strong

dependence between the model parameters and the auxiliary variables, with only a fraction of the auxiliary variables allowed to be updated in a given iteration.

3. UNBIASED LIKELIHOOD ESTIMATOR

3.1. Unbiased and positive likelihood estimators with control variates. The Bardenet et al. (2015) strategy is to first construct a sequence of unbiased estimators of the log-likelihood and then obtain a positive and unbiased estimate of the likelihood $p(y|\theta)$ by the debiasing technique in Rhee and Glynn (2015); see also Strathmann et al. (2015) for an alternative use of debiasing to estimate posterior expectations by combining estimates from a sequence of partial posteriors. To ensure that the debiased estimate is positive, the unbiased log-likelihood estimators in the sequence must have a lower bound $a(\theta)$ (Jacob and Thiery, 2015). Assume that the likelihood decomposes as

$$(3.1) \quad p(y|\theta) = \exp(l(\theta)), \quad l(\theta) = \sum_{k=1}^n l_k(\theta), \quad \text{where } l_k(\theta) = \log p(y_k|\theta, x_k)$$

is the log-likelihood contribution of the k th observation. To reduce the variance of the likelihood estimator in Bardenet et al. (2015) we introduce a control variate $q_k(\theta)$ for the k th observation such that $q_k(\theta) \approx l_k(\theta)$. Notice that (3.1) can be rewritten as

$$p(y|\theta) = \exp(l(\theta)), \quad l(\theta) = q(\theta) + \sum_{k=1}^n [l_k(\theta) - q_k(\theta)], \quad \text{where } q(\theta) = \sum_{k=1}^n q_k(\theta).$$

From now on we often suppress dependence on θ to simplify notation. Define

$$(3.2) \quad d_k = l_k - q_k, \quad d = \sum_{k=1}^n d_k$$

and we now define two unbiased estimators of d based on different sampling schemes. The first estimator has a fixed sample size m_1 and is defined as

$$(3.3) \quad \hat{d}_{m_1} = \frac{n}{m_1} \sum_{i=1}^{m_1} d_{u_i}, \quad \text{with } \Pr(u_i = k) = \frac{1}{n}, \quad k = 1, \dots, n, \quad \text{and the } u_i\text{'s are iid.}$$

The second estimator is defined using binary u_k , $k = 1, \dots, n$, and an expected sample size $m_2 = E[\sum_{k=1}^n u_k]$,

$$(3.4) \quad \hat{d}_{m_2} = \frac{n}{m_2} \sum_{k=1}^n u_k d_k, \text{ with } \Pr(u_k = 1) = \frac{m_2}{n}, \quad k = 1, \dots, n, \text{ and the } u_k \text{'s are iid.}$$

To construct an unbiased likelihood estimator, consider a sequence of $h = 1, \dots, G$, estimators $\hat{d}_{m_b}^{(h)}$ as in (3.3) or (3.4), each based on its own subset of observations $u^{(h)}$ with m_b elements. Hence the subscript b indicates that it considers a *batch* of observations and consequently the variance for a single batch is denoted by $\sigma_b^2 = V[\hat{d}_{m_b}]$. Let $a(\theta)$ be a lower bound for all $\hat{d}_m^{(h)}$. The debiasing estimator in Rhee and Glynn (2015) (RG) considered by Bardenet et al. (2015), which is a positive unbiased estimator of $L(\theta) = p(y|\theta)$ based on a prior expected sample size \bar{m} , is

$$\hat{L}_{\bar{m}}^{\text{RG}} = \exp(q + a) \left(1 + \sum_{g=1}^G \frac{1}{\Pr(G \geq g)} \frac{1}{g!} \prod_{h=1}^g (\hat{d}_{m_b}^{(h)} - a) \right), \quad \bar{m} = m_b E[G].$$

It can be shown that $E[\hat{L}_{\bar{m}}(\theta)] = L(\theta)$ for any θ and any distribution for G . We follow Bardenet et al. (2015) and use

$$(3.5) \quad G \sim \text{Geometric}(p) \quad \text{with } p = \epsilon/(1 + \epsilon), \quad \epsilon > 0,$$

with resulting estimator

$$(3.6) \quad \hat{L}_{\bar{m}}^{\text{RG}} = \exp(q + a) \left(1 + \sum_{g=1}^G \frac{(1 + \epsilon)^g}{g!} \prod_{h=1}^g (\hat{d}_{m_b}^{(h)} - a) \right), \quad \bar{m} = m_b/p.$$

Alternatively, we can use the Poisson (Pois) estimator

$$(3.7) \quad \hat{L}_{\bar{m}}^{\text{Pois}} = \exp(q + a + \lambda) \prod_{h=1}^G \left(\frac{\hat{d}_{m_b}^{(h)} - a}{\lambda} \right), \quad G \sim \text{Poisson}(\lambda), \quad \bar{m} = m_b \lambda,$$

which is also unbiased for $L(\theta)$ and positive (if a is a lower bound). If $G = 0$ the product is defined to be 1. For both estimators, we refer to $\bar{m} = m_b E[G]$ as the prior expected subsample size because the actual computational cost of the estimator will depend on the

realized path of G , see Section 4.5. In addition, there is a cost for the control variates which is considered in Section 3.2.

The variance of $\hat{L}_{\bar{m}}^{\text{RG}}$ in (3.6) and also $V[\log \hat{L}_{\bar{m}}^{\text{RG}}]$ are difficult to compute. In contrast, we can derive the following results for the Poisson estimator.

Lemma 1.

i. *The variance of the Poisson likelihood estimator in (3.7) is*

$$V[\hat{L}_{\bar{m}}^{\text{Pois}}(\theta)] = \exp(2(q+a)) \left(\exp\left(\lambda + \frac{1}{\lambda}(\sigma_b^2 + (d-a)^2)\right) - \exp(2(d-a)) \right)$$

ii. *The variance of the Log-Likelihood (LL) estimator $V[\log \hat{L}_{\bar{m}}^{\text{Pois}}(\theta)]$ is approximately*

$$\sigma_{LL}^2 = \lambda \frac{\sigma_b^2}{(d-a)^2} + \lambda \left(\log\left(\frac{d-a}{\lambda}\right) - \frac{\sigma_b^2}{2(d-a)^2} \right)^2.$$

Proof. Part (i) is derived using Equation (21) in Papaspiliopoulos (2009). Part (ii) is proved in Appendix A. \square

Proposition 4.1 in Bardenet et al. (2015) derives a lower bound for the relative variance of (3.6). Bardenet et al. (2015) conclude that the number of observations used for estimation needs to grow impractically large as a function of n to keep this variance bounded. Our methodology has two important mechanisms to avoid this behavior. First, the quality of the approximations underlying our control variates in Section 3.2 improves as n grows (Quiroz et al., 2016). Second, the correlated pseudo-marginal approach in Section 4 makes it possible to have a much larger variance than in the uncorrelated case (Quiroz et al., 2016).

Proposition 1 in Papaspiliopoulos (2009) states that the optimal implementation (in terms of minimizing $V[\hat{L}_{\bar{m}}^{\text{Pois}}(\theta)]$) is achieved when $a = -\lambda$ and $\lambda \rightarrow \infty$, in which case $V[\hat{L}_{\bar{m}}^{\text{Pois}}(\theta)]$ approaches 0, but this is not practical because our goal is to speed up computations. For a fixed value a it follows that $\lambda = \sqrt{\sigma_b^2 + (d-a)^2}$ is optimal. On the other hand, for a fixed λ , $a = d - \lambda$ is optimal. The latter observation gives an interesting comparison to the log-likelihood estimator in Quiroz et al. (2016). When implementing their estimator of

the log-likelihood with a sample size $m = m_b\lambda$, one can show that their variance of the log-likelihood estimator can be expressed as $\tilde{\sigma}_{LL}^2 = \sigma_b^2/\lambda$. By using the optimality condition $\lambda = d - a$ it follows that

$$\begin{aligned}\sigma_{LL}^2 &= \lambda \frac{\sigma_b^2}{(d-a)^2} + \lambda \left(\log \left(\frac{d-a}{\lambda} \right) - \frac{\sigma_b^2}{2(d-a)^2} \right)^2 \\ &= \frac{\sigma_b^2}{\lambda} + \frac{\sigma_b^4}{4\lambda^3} \\ &> \tilde{\sigma}_{LL}^2,\end{aligned}$$

i.e. the variance of the log of the Poisson estimator is always larger in this setting. Hence the exactness comes at the cost of increasing the variance compared to Quiroz et al. (2016), which explains why the approximate subsampling MCMC in Quiroz et al. (2016) is the preferred choice if computational cost is the primary concern.

3.2. Control variates. We follow Quiroz et al. (2016) and obtain a sparse set of the data by clustering the data into K clusters before the MCMC. Within a cluster, we set $q_k(\theta)$ to a Taylor series approximation of $l_k(\theta)$ around the cluster centroid. This allows us to compute q in (3.6) by K operations rather than n ; see Quiroz et al. (2016) for details. The K operations are added to the computational cost, so that in total our estimators use $G \times m_b + K$ evaluations, and the prior expected number of evaluations is $\bar{m} + K$.

It is also possible to instead use the control variates suggested in Bardenet et al. (2015), expanding w.r.t θ (instead of the data) around some reference value θ^* .

3.3. Soft lower bound. While the lower bound of all $\hat{d}_m^{(h)}$ ensures that (3.6) is positive, it is not practical for two reasons. First, for most models we will typically need to evaluate $d_k = l_k - q_k$ for all data points to find a lower bound. Second, since the lower bound needs to apply to all data points, it will necessarily be much too low for most realized values of u . If the lower bound becomes too negative this will inflate the variance of (3.6) and (3.7), where we can use Lemma 1 to verify this for the latter. We therefore advocate using a *soft lower bound* \tilde{a} , which we define to be a lower bound that holds with a probability \tilde{p} for a realization of u conditional on a value of G . However, the estimator can now be negative

and we outline how to carry out MCMC in Section 4.1. We now turn to how to construct a soft lower bound \tilde{a} for a given \tilde{p} .

For brevity we only treat the case of a fixed subsample m_b within a batch as in (3.3). A random subsample size is treated analogously. We assume that

$$(3.8) \quad \hat{d}_{m_b}^{(h)} \sim N(d, \sigma_b^2), \quad \sigma_b^2 = \frac{n}{m_b} \sum_{k=1}^n (d_k - \bar{d})^2, \quad \text{for } h = 1, \dots, G,$$

which is justified by the standard central limit theorem because the observations are iid. We then want to find an \tilde{a} such that

$$(3.9) \quad \Pr(\hat{L}_{\bar{m}} > 0) \geq \tilde{p} = \Pr \left[\min \left(\hat{d}_m^{(1)}, \hat{d}_m^{(2)}, \dots, \hat{d}_m^{(G)} \right) > \tilde{a} \right],$$

where \tilde{p} is close to 1 and specified by the user. Note that although $\Pr(\hat{L}_{\bar{m}} > 0)$ is intractable for both (3.6) and (3.7) it increases with \tilde{p} . It is straightforward to show that if σ_b^2 is known, then

$$\tilde{p} = \left[1 - \Phi \left(\frac{\tilde{a} - d}{\sigma_b} \right) \right]^G,$$

where Φ is the standard normal cdf function. Therefore,

$$\tilde{a} = d + \sigma_b \Phi^{-1} (1 - \tilde{p}^{1/G})$$

is a lower bound with probability \tilde{p} . This requires the full data set so we instead use \hat{d} and $\hat{\sigma}_b$ in place of d and σ_b , and thus change Φ^{-1} to a Student- t inverse cdf with $m - 1$ degrees of freedom. During the burn-in period we take \tilde{a} to depend on u , and subsequently set \tilde{a} to a constant equal to the average over the burn-in draws. This \tilde{a} is plugged into (3.6) or (3.7) and defines our final (possibly negative) likelihood estimators. Fixing \tilde{a} ensures that the estimators are unbiased and draws used for learning \tilde{a} are discarded from the final MCMC sample. Note that since \tilde{a} is trained on the same subset as the one used for estimating the likelihood it comes with no additional computational cost. Alternatively, we can before the MCMC compute the true d and σ at some central value of θ , say the mode, at the cost of computing on the full data set once.

4. METHODOLOGY

4.1. A pseudo-marginal algorithm with a possibly negative estimator. Let $p(\theta)$ and $\pi(\theta) = p(\theta|y)$ denote the prior and the posterior for θ , respectively. Let $p(u, G)$ be the joint distribution of the vector $u = u^{(1)}, \dots, u^{(G)}$, where each $u^{(h)}$ is a vector of auxiliary variables corresponding to the subset of observations to include in batch h when estimating $p(y|\theta)$. The u 's are observation indices for the estimator in (3.3), and binary for the estimator in (3.4).

Let $\hat{p}_{\bar{m}}(y|\theta, u, G) = \hat{L}_{\bar{m}}(\theta)$ denote any of the unbiased estimators of $p(y|\theta)$ in (3.6) or (3.7). The unbiasedness condition means that

$$(4.1) \quad p(y|\theta) = \int_G \int_u \hat{p}_{\bar{m}}(y|\theta, u, G) p(u, G) du dG.$$

Define

$$(4.2) \quad \tilde{\pi}(\theta, u, G) = \hat{p}_{\bar{m}}(y|\theta, u, G) p(u, G) p(\theta) / p(y), \text{ with } p(y) = \int p(y|\theta) p(\theta) d\theta,$$

on the augmented space (θ, u, G) . Although $\tilde{\pi}(\theta, u, G)$ integrates to 1, it is not a proper distribution because $\hat{p}_{\bar{m}}(y|\theta, u, G)$ may be negative because of our soft lower bound, and therefore the pseudo-marginal algorithm (Beaumont, 2003; Andrieu and Roberts, 2009) does not apply directly. Lyne et al. (2015) target the absolute measure of the target posterior with a pseudo-marginal algorithm and show that by properly weighting these iterates it is possible to get a consistent estimate of $E_{\pi}(h) := \int h(\theta) \pi(\theta) d\theta$ for some function $h(\theta)$. This elegant observation allows exact inference in the sense that $h(\theta)$ is indeed averaged over the true target. We now outline the MCMC scheme in Lyne et al. (2015) applied to our setting.

The joint proposal is given by

$$(4.3) \quad q(\theta, u, G | \theta_c, u_c, G_c) = q(\theta | \theta_c) p(u | G) p(G)$$

where the subscript c denotes the current state of the Markov chain. Now propose $\theta_p, u_p, G_p \sim q(\theta, u, G | \theta_c, u_c, G_c)$ and compute

$$(4.4) \quad \alpha = \min \left(1, \frac{|\hat{p}_{\bar{m}}(y | \theta_p, u_p, G_p)| p(\theta_p) / q(\theta_p | \theta_c)}{|\hat{p}_{\bar{m}}(y | \theta_c, u_c, G_c)| p(\theta_c) / q(\theta_c | \theta_p)} \right),$$

set

$$\{\theta^{(i)}, u^{(i)}, G^{(i)}\} = \begin{cases} \{\theta_p, u_p, G_p\} & \text{if accepted} \\ \{\theta_c, u_c, G_c\} & \text{if rejected,} \end{cases}$$

and record $s(\theta^{(i)}, u^{(i)}, G^{(i)}) = \text{sign}(\hat{p}_{\bar{m}}(y | \theta^{(i)}, u^{(i)}, G^{(i)}))$. Lyne et al. (2015) propose using these iterates to form the importance sampling estimate

$$(4.5) \quad \hat{E}_\pi(h) = \frac{\sum_{i=1}^N h(\theta^{(i)}) s(\theta^{(i)}, u^{(i)}, G^{(i)})}{\sum_{i=1}^N s(\theta^{(i)}, u^{(i)}, G^{(i)})},$$

and prove that it is a consistent estimator. We note that the variance of $\hat{E}_\pi(h)$ is inflated if too many iterates with a negative sign are accepted. Our scheme is efficient if (i) the pseudo-marginal is efficient and (ii) the variance of (4.5) is small. The efficiency of pseudo-marginal is achieved by variance reduction through the control variates (Section 3.2), correlating subsamples (Section 4.3 and 4.4), and finally not making the lower bound too extreme (Section 3.3). The variance of the importance sampling estimate is by construction low (given that the pseudo-marginal is efficient), as we choose \tilde{p} close to 1 and therefore $\Pr(\hat{L}_{\bar{m}} < 0)$ is small.

4.2. Correlated pseudo-marginal MCMC. The efficiency of the pseudo-marginal chain depends crucially on the level of the variance $\sigma_{LL}^2 = \text{V}[\log \hat{p}_{\bar{m}}(y | \theta, u, G)]$. In particular, a large variance can easily produce extreme over-estimates of the likelihood and cause the Markov chain to get stuck for long spells. On the other hand, a too small σ_{LL}^2 is not desirable as it becomes prohibitively expensive to compute the estimator. Pitt et al. (2012), Doucet et al. (2015) and Sherlock et al. (2015) analyze the value of σ_{LL}^2 that optimizes the trade off between efficiency and computational time. The optimal value of σ_{LL}^2 ranges between 1 and 3.3, where larger values are tolerated when the proposal for θ is inefficient. Recent advances

in the pseudo-marginal MCMC literature correlates the random numbers underlying the estimator at the proposed (numerator) and current (denominator) in (4.4). The so called correlated pseudo-marginal method in Deligiannidis et al. (2016) (see also Dahlin et al., 2015) is a tremendous gain for pseudo-marginal algorithms because we can target $\sigma_{LL}^2 \gg 1$. As a consequence we can use much less precise estimators without losing any efficiency of the pseudo-marginal chain.

Quiroz et al. (2016) introduce the idea of correlating the estimators in a subsampling setting. However, as described in Section 2, the error in the bias corrected estimator depends explicitly on the level of σ_{LL}^2 , and it might be the case that it is not possible to target a too large σ^2 . The estimators proposed here are unbiased (without any correction) and can target a large σ_{LL}^2 without implications for the exactness of the method. We now propose two ways of correlating the estimators in (4.4).

4.3. Correlating G . We can think of the vector u as a composite of G blocks, where the set of random variables $u^{(h)}$ for block h are used to form the unbiased estimate $\hat{d}_{mb}^{(h)}$ of $d(\theta)$ in (3.2). Recall that the collection of these G estimators is used to form (3.6) or (3.7), which in turn are used to estimate the MH ratio in (4.4). The pseudo-marginal in the previous section uses an independent set of u, G at the numerator and denominator for estimating the likelihood. We now propose to make the set of random numbers u_c, G_p and u_p, G_p dependent by making G_c and G_p positively correlated. Intuitively, if G is large, the correlation between estimates in the MH ratio is high, because the random numbers will only differ for a relatively small number of blocks.

We correlate G_c and G_p by a copula transformation with underlying normal variates as in Deligiannidis et al. (2016). Specifically, we propose

$$(4.6) \quad v_p = \phi v_c + \sqrt{1 - \phi^2} \xi, \quad \xi \sim \mathcal{N}(0, 1),$$

set $w_p = \Phi(v_p) \sim \text{Unif}(0, 1)$, and compute $G_p = F^{-1}(w_p)$, where F^{-1} is the inverse cdf of the geometric distribution in (3.5), or the Poisson distribution in (3.7). This induces a correlation between G_c and G_p which is a deterministic function of the v 's, so the theory

in Deligiannidis et al. (2016) applies. Note that, in turn, the correlation between G_c and G_p induces a correlation between u_c (observation indices included in current draw) and u_p (observations indices included in proposed draw). If $G_p > G_c$ we add $G_p - G_c$ blocks of independent u 's, and if the opposite we delete $G_c - G_p$ blocks at random. Setting ϕ close to 1 gives a strong positive correlation between G_c and G_p which are then expected to be close. This way of inducing the correlation is similar to Tran et al. (2016), although the interpretation of the correlation in terms of G is highly complex in our setting, especially for the estimator in (3.6).

4.4. Correlating u within a block. We note that when $G_p = G_c$ in Section 4.3 the u 's do not move. To explore a faster mixing of the u 's we can in addition also correlate within each block. We follow Quiroz et al. (2016) and consider the case of binary u with a random sample size. For each block that is not removed or added, we generate u_p from a Markov chain with marginal $p(u_c = 1) = m_b/n$ (m_b is the expected batch size), with transition probabilities $\Pr(u_p = 1|u_c = 1) = \kappa$ and $\Pr(u_p = 0|u_c = 0) = 1 - (1 - \kappa) \frac{m_b/n}{1 - m_b/n}$. Quiroz et al. (2016) show that this is equivalent to generating normal variates as in (4.6) and therefore the theory in Deligiannidis et al. (2016) applies directly.

4.5. Evaluating the efficiency. In Section 3.1 we introduced the (expected) prior computational cost as $\bar{m} = E[G]m$. We measure the *realized* computational cost by $\bar{m}_r = E[G_p|G_c]m$, where $E[G_p|G_c]$ is the expected value of the proposed G conditional on the accepted G_c . This is easily estimated as the mean of the proposed values of G . We take as efficiency measure the Effective Draws (ED)

$$(4.7) \quad \text{ED} = \frac{\text{IF}}{c}, \quad \text{Inefficiency Factor (IF)} = 1 + 2 \sum_{l=1}^{\infty} \rho_l,$$

and $c = \bar{m}_r + K$ is (proportional to) the computational cost, where K is the number of clusters in Section 3.2.

5. EXPERIMENTS

5.1. Models. We implement both uncorrelated and correlated (G and $G + u$) PMMH with the estimators in (3.6) and (3.7). We compare our algorithms against standard MH and Firefly Monte Carlo (Maclaurin and Adams, 2014), the only previously proposed exact subsampling MCMC algorithm. Our experiments use the autoregressive time series models in Quiroz et al. (2016) when comparing to other subsampling approaches. The models are two AR(1) processes with iid Student- t errors $\epsilon_t \sim t(\nu)$ with known degrees of freedom ν . The two data generating processes are

$$(5.1) \quad y_t = \begin{cases} \beta_0 + \beta_1 y_{t-1} + \epsilon_t & , [\mathbf{M}_1, \theta = (\beta_0 = 0.3, \beta_1 = 0.6)] \\ \mu + \rho(y_{t-1} - \mu) + \epsilon_t & , [\mathbf{M}_2, \theta = (\mu = 0.3, \rho = 0.99)] \end{cases},$$

where $p(\epsilon_t | \theta) \propto (1 + \epsilon_t^2 / \nu)^{-(\nu+1)/2}$ with $\nu = 5$ and uniform priors

$$p(\beta_0, \beta_1) \stackrel{\text{ind.}}{=} \mathcal{U}(-5, 5) \cdot \mathcal{U}(0, 1) \quad \text{and} \quad p(\mu, \rho) \stackrel{\text{ind.}}{=} \mathcal{U}(-5, 5) \cdot \mathcal{U}(0, 1).$$

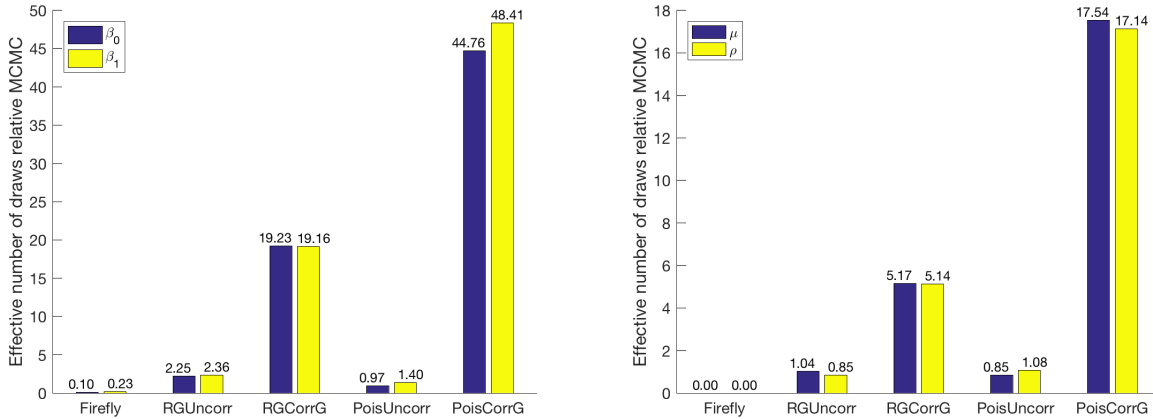


FIGURE 1. Effective draws in (4.7) relative to standard MH for model \mathbf{M}_1 (left panel) and \mathbf{M}_2 (right panel).

5.2. Experimental settings. For the correlated algorithms we use $E[G] = 50$ which gives $\epsilon \approx 0.0204$ for (3.6) and $\lambda = 50$ for (3.7). This value is practically large enough to avoid the adverse effect a small G has on the efficiency of the Markov chain when correlating:

the noise in the MH ratio can be severe due to a relatively large fraction of the u_p vector being independent of u_c . For the uncorrelated algorithm we set ϵ and λ corresponding to $E[G] = 5$. For both estimators we set $\phi = 0.9999$ when correlating G and $\kappa = 0.9863$ when correlating u within a block. Quiroz et al. (2016) show that this κ corresponds to the persistence parameter $\phi = 0.9999$.

We generate $n = 100,000$ observations following the data generating processes in (5.1). The number of clusters used for the control variates are $K = 1\%$ for M_1 (in % of n) and $K = 3.2\%$, for M_2 . We use a Random Walk Metropolis proposal $q(\theta|\theta_c) = \mathcal{N}(\theta_c, l\Sigma_{\theta^*})$, where Σ_{θ^*} is the covariance matrix at the posterior mode θ^* , and l is adapted during the burn-in phase to reach an acceptance probability of $\alpha = 0.35$ for the standard MH (Gelman et al., 1996). Likewise, the pseudo-marginal algorithms are adapted to reach an acceptance probability of $\alpha = 0.15$ and, moreover, target $\sigma_{LL}^2 \approx 2.1$ as the five parameter example in Sherlock et al. (2015) for the uncorrelated version. For the correlated version we experiment on how large σ_{LL}^2 we can target and conclude that $\sigma^2 \approx 400$ is reasonable. Note that the unbiased estimator in (3.6) has an intractable σ_{LL}^2 , and therefore we compute it by Monte Carlo simulation with $\theta = \theta^*$. For (3.7) we instead use part (ii) of Lemma 1. For Firefly Monte Carlo we set the re-sampling fraction to 10% and implement the lower bound using Taylor series proxies as suggested by Bardenet et al. (2015).

5.3. Comparisons against standard MH and Firefly Monte Carlo. Figure 1 shows the number of effective draws relative to standard MH for our proposed algorithms and Firefly. To save space we have not included the case when correlating both u and G , as the results are similar to only correlating G . This indicates a fast mixing of the auxiliary variables. The figure shows that correlating the estimators as in Section 4.3 and 4.4 is crucial for obtaining significant gains over standard MH. The correlation allows us to target a much higher variance and therefore use a much smaller sample for estimation, see Table 1. The control variates are necessary for success: only correlating will not be sufficient to prevent the sampler from getting stuck (not shown here). It is evident that the Poisson estimator

is more efficient than the RG estimator, which is expected because the latter has a highly complex structure.

TABLE 1. Mean of sampling fraction (\bar{m}_r/n for our algorithms, which includes K) over MCMC iterations for models M_1 and M_2 with: standard MH (MCMC), Firefly Monte Carlo (Firefly), uncorrelated pseudo-marginal (Uncorr) and correlated G pseudo-marginal (correlating G or $G + u$). The table shows the result for both estimators in (3.6) (RG) and (3.7) (Poisson).

	MCMC	Firefly	Uncorr		Corr G		Corr $G + u$	
			RG	Poisson	RG	Poisson	RG	Poisson
M_1	1.000	0.101	0.202	0.060	0.033	0.014	0.036	0.013
M_2	1.000	0.196	0.435	0.182	0.124	0.037	0.116	0.037

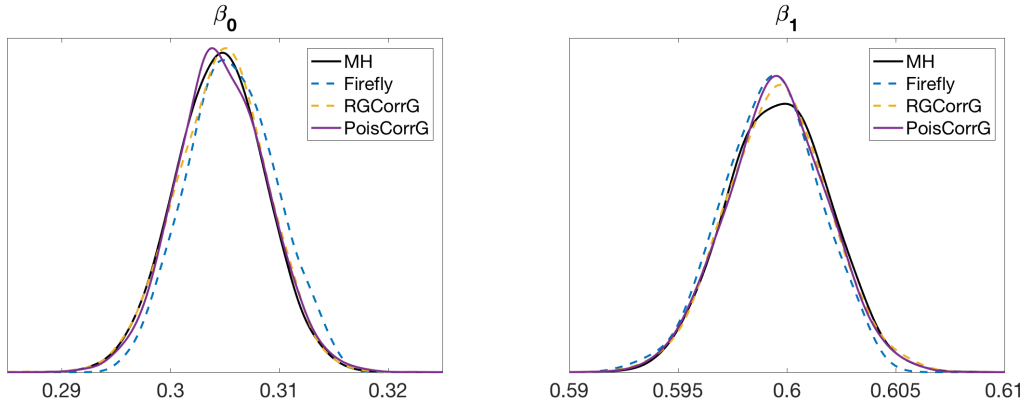
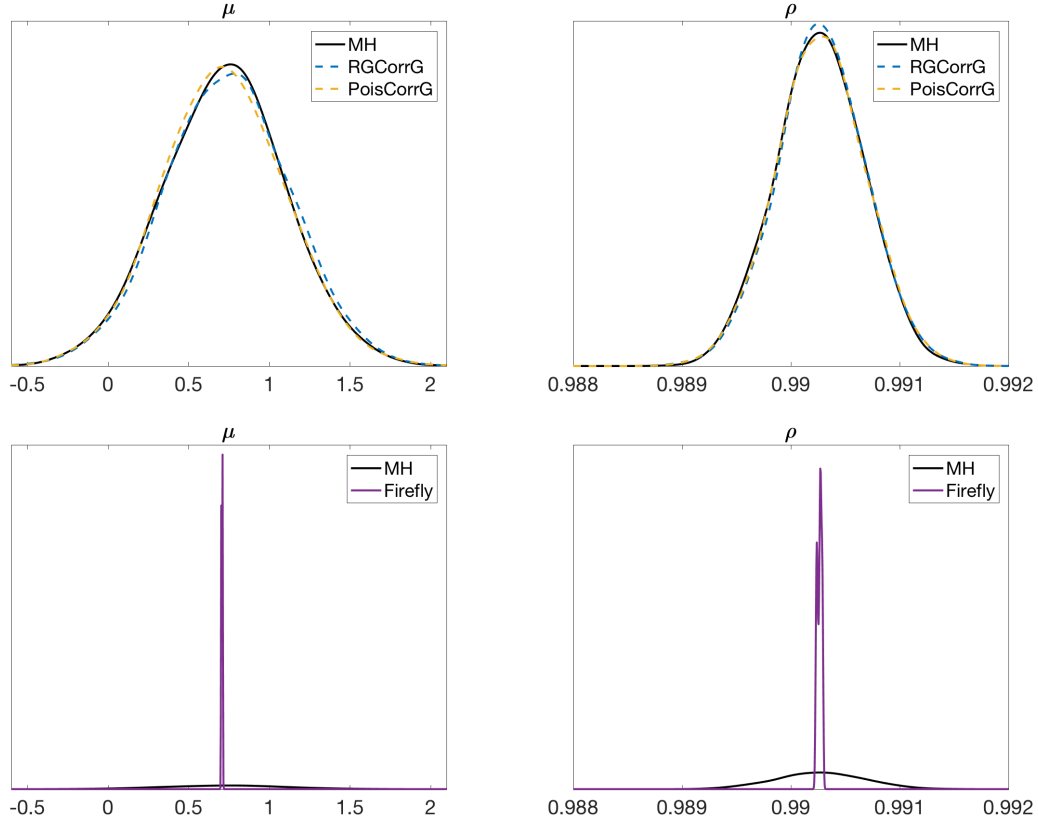


FIGURE 2. Kernel density estimates for the two parameters in M_1 .

Figures 2 and 3 show kernel density estimates based on the posterior draws for the compared algorithms (again excluding $G + u$ correlation because it gives similar results as only correlating G). Note that while these are draws from a perturbed posterior which is obtained by integrating the absolute measure with respect to u and G , they are representative of the posterior since very few draws had negative likelihood estimates. This is verified in Table 2, which shows that there is essentially no difference between posterior quantiles estimated directly from draws from this perturbed posterior and posterior quantiles estimated from the correct importance sampling estimator in (4.5). Figure 3 also shows that although Firefly Monte Carlo is theoretically an exact algorithm, it can be far from the true posterior in finite samples, at least in this example.

FIGURE 3. Kernel density estimates for the two parameters in M_2 .

6. CONCLUSIONS AND FUTURE RESEARCH

We propose an algorithm for exact inference by data subsampling. The key features of our approach are (i) efficient control variates for variance reduction and (ii) variance reduction of the estimated MH ratio via a correlated pseudo-marginal approach and (iii) the use of a soft lower bound in the occasionally negative likelihood estimator.

We perform experiments on two illustrative models and document the following findings. First, our algorithm clearly outperforms the standard MH, especially when estimating the likelihood with the Poisson estimator. Second, we dramatically outperform Firefly Monte Carlo, which is the only previously proposed exact subsampling algorithm in the literature. To illustrate the magnitude of improvements, our algorithms that correlate give for each example, respectively, between 18-52 and 5-18 more effective draws than MCMC on the full data using a measure that balances execution cost and efficiency of the resulting chain. The

TABLE 2. *Quantile estimation for parameter μ in model M_2 .* The results are divided into the estimator in (3.6) (RG) and (3.7) (Poisson). Estimation of $\alpha = \Pr(\theta \leq c_\alpha)$ by the Monte Carlo Estimate (MCE) $\frac{1}{N} \sum_{i=1}^N h(\theta^{(i)})$ with $h(\theta) = \mathbb{I}(\theta \leq c_\alpha)$ and θ drawn from the perturbed posterior (after integrating the absolute measure with respect to u and G), and the Importance Sampling Estimate (ISE) in (4.5). The quantiles are obtained from the standard MH algorithm. The same accuracy also holds for ρ and for the parameters in M_1 .

	$\alpha = 0.10$		$\alpha = 0.25$		$\alpha = 0.50$		$\alpha = 0.75$		$\alpha = 0.90$	
	MCE	ISE	MCE	ISE	MCE	ISE	MCE	ISE	MCE	ISE
R-G										
Uncorr	0.092	0.092	0.240	0.240	0.488	0.487	0.737	0.737	0.905	0.905
Corr G	0.089	0.089	0.243	0.242	0.481	0.482	0.737	0.737	0.887	0.887
Corr $G + u$	0.098	0.098	0.260	0.261	0.522	0.522	0.747	0.747	0.903	0.903
Poisson										
Uncorr	0.100	0.100	0.248	0.248	0.484	0.484	0.716	0.715	0.875	0.875
Corr G	0.095	0.095	0.258	0.259	0.516	0.516	0.755	0.755	0.902	0.902
Corr $G + u$	0.105	0.105	0.255	0.255	0.503	0.503	0.742	0.743	0.897	0.897

corresponding numbers when comparing against the Firefly MC algorithm are 78-530 and 3,573-11,983.

Future research should explore how $E[G]$ relates to the correlation between the logarithm of the estimators in (3.6) or (3.7) at the proposed and current state of the Markov chain. One could then choose $E[G]$ in order to achieve a high correlation. Moreover, it is of interest to derive the theoretical inefficiency and also $\Pr(\hat{L}_{\tilde{m}} > 0)$ as a function of the soft lower bound \tilde{a} and σ_b^2 , for a given $E[G]$. We can then choose \tilde{a} and m_b to minimize the variance of the importance sampling step derived in Lyne et al. (2015), while also taking into account the computational cost. This gives us a sensible way to choose the tuning parameters in our algorithm. We remark that the Poisson estimator is important, not only because it significantly improves the empirical results, but also makes these derivations possible (in contrast to (3.6) which is intractable for this purpose).

REFERENCES

- Andrieu, C. and Roberts, G. O. (2009). The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, pages 697–725.
- Banterle, M., Grazian, C., and Robert, C. P. (2014). Accelerating Metropolis-Hastings algorithms: Delayed acceptance with prefetching. *arXiv preprint arXiv:1406.2660*.
- Bardenet, R., Doucet, A., and Holmes, C. (2014). Towards scaling up Markov chain Monte Carlo: an adaptive subsampling approach. In *Proceedings of The 31st International Conference on Machine Learning*, pages 405–413.
- Bardenet, R., Doucet, A., and Holmes, C. (2015). On Markov chain Monte Carlo methods for tall data. *arXiv preprint arXiv:1505.02827*.
- Beaumont, M. A. (2003). Estimation of population growth or decline in genetically monitored populations. *Genetics*, 164(3):1139–1160.
- Beskos, A., Papaspiliopoulos, O., Roberts, G. O., and Fearnhead, P. (2006). Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):333–382.
- Christen, J. A. and Fox, C. (2005). MCMC using an approximation. *Journal of Computational and Graphical Statistics*, 14(4):795–810.
- Dahlin, J., Lindsten, F., Kronander, J., and Schön, T. B. (2015). Accelerating pseudo-marginal Metropolis-Hastings by correlating auxiliary variables. *arXiv preprint arXiv:1511.05483*.
- Deligiannidis, G., Doucet, A., and Pitt, M. K. (2016). The correlated pseudo-marginal method. *arXiv preprint arXiv:1511.04992v3*.
- Doucet, A., Pitt, M., Deligiannidis, G., and Kohn, R. (2015). Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator. *Biometrika*, page asu075.
- Fearnhead, P., Papaspiliopoulos, O., Roberts, G. O., and Stuart, A. (2010). Random-weight particle filtering of continuous time processes. *Journal of the Royal Statistical Society:*

- Series B (Statistical Methodology)*, 72(4):497–512.
- Gelman, A., Roberts, G., and Gilks, W. (1996). Efficient Metropolis jumping rules. *Bayesian statistics*, 5(599-608):42.
- Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109.
- Jacob, P. E. and Thiery, A. H. (2015). On nonnegative unbiased estimators. *The Annals of Statistics*, 43(2):769–784.
- Korattikara, A., Chen, Y., and Welling, M. (2014). Austerity in MCMC land: Cutting the Metropolis-Hastings budget. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 181–189.
- Liu, S., Mingas, G., and Bouganis, C.-S. (2015). An exact MCMC accelerator under custom precision regimes. In *Field Programmable Technology (FPT), 2015 International Conference on*, pages 120–127. IEEE.
- Lyne, A.-M., Girolami, M., Atchade, Y., Strathmann, H., and Simpson, D. (2015). On Russian roulette estimates for Bayesian inference with doubly-intractable likelihoods. *Statistical Science*, 30(4):443–467.
- Maclaurin, D. and Adams, R. P. (2014). Firefly Monte Carlo: Exact MCMC with subsets of data. In *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence (UAI 2014)*.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092.
- Papaspiliopoulos, O. (2009). A methodological framework for Monte Carlo probabilistic inference for diffusion processes.
- Payne, R. D. and Mallick, B. K. (2015). Bayesian big data classification: A review with complements. *arXiv preprint arXiv:1411.5653v2*.
- Pitt, M. K., Silva, R. d. S., Giordani, P., and Kohn, R. (2012). On some properties of Markov chain Monte Carlo simulation methods based on the particle filter. *Journal of*

- Econometrics*, 171(2):134–151.
- Quiroz, M. (2016). Speeding up MCMC by delayed acceptance and data subsampling. *arXiv preprint arXiv:1507.06110v2*.
- Quiroz, M., Villani, M., Kohn, R., and Tran, M.-N. (2016). Speeding up MCMC by efficient data subsampling. *arXiv preprint arXiv:1404.4178v4*.
- Rhee, C. and Glynn, P. W. (2015). Unbiased estimation with square root convergence for SDE models. *Operations Research*, 63(5):1026–1043.
- Sherlock, C., Thiery, A. H., Roberts, G. O., and Rosenthal, J. S. (2015). On the efficiency of pseudo-marginal random walk Metropolis algorithms. *The Annals of Statistics*, 43(1):238–275.
- Strathmann, H., Sejdinovic, D., and Girolami, M. (2015). Unbiased Bayes for big data: Paths of partial posteriors. *arXiv preprint arXiv:1501.03326*.
- Tran, M.-N., Kohn, R., Quiroz, M., and Villani, M. (2016). Block-wise pseudo-marginal Metropolis-Hastings. *arXiv preprint arXiv:1603.02485v2*.
- Wagner, W. (1988). Unbiased multi-step estimators for the Monte Carlo evaluation of certain functional integrals. *Journal of Computational Physics*, 79(2):336–352.
- Wagner, W. (1989). Unbiased Monte Carlo estimators for functionals of weak solutions of stochastic differential equations. *Stochastics: An International Journal of Probability and Stochastic Processes*, 28(1):1–20.

APPENDIX A. PROOF OF LEMMA 1

Proof of part (ii) of Lemma 1. By the law of total variance

$$\mathrm{V} \left[\log \hat{L}_{\bar{m}}^{\mathrm{Pois}}(\theta) \right] = \mathrm{E}_G \left[\mathrm{V}_{u|G} \left[\log \hat{L}_{\bar{m}}^{\mathrm{Pois}}(\theta) \middle| G \right] \right] + \mathrm{V}_G \left[\mathrm{E}_{u|G} \left[\log \hat{L}_{\bar{m}}^{\mathrm{Pois}}(\theta) \middle| G \right] \right].$$

We use the delta-method and obtain the following approximations (in a neighborhood of $d - a$)

$$\begin{aligned} V \left[\log \left(\hat{d}_{m_b}^{(h)} - a \right) \right] &\approx \left(\frac{1}{d - a} \right)^2 V \left[\hat{d}_{m_b}^{(h)} \right] = \frac{\sigma_b^2}{(d - a)^2} \\ E \left[\log \left(\hat{d}_{m_b}^{(h)} - a \right) \right] &\approx \log(d - a) - \frac{1}{2(d - a)^2} \sigma_b^2. \end{aligned}$$

With these approximations we first obtain

$$\begin{aligned} V_{u|G} \left[\log \hat{L}_{\bar{m}}^{\text{Pois}}(\theta) \middle| G \right] &= \sum_{h=1}^G V \left[\log \left(\hat{d}_{m_b}^{(h)} - a \right) \right] \\ &\approx G \frac{\sigma_b^2}{(d - a)^2}, \end{aligned}$$

and taking the outer expectation

$$E_G \left[V_{u|G} \left[\hat{L}_{\bar{m}}^{\text{Pois}}(\theta) \middle| G \right] \right] \approx \lambda \frac{\sigma_b^2}{(d - a)^2}.$$

Next,

$$\begin{aligned} E_{u|G} \left[\hat{L}_{\bar{m}}^{\text{Pois}}(\theta) \middle| G \right] &= q + a + \lambda - G \log(\lambda) + \sum_{h=1}^G E \left[\log \left(\hat{d}_{m_b}^{(h)} - a \right) \right] \\ &\approx q + a + \lambda - G \log(\lambda) + G \left(\log(d - a) - \frac{1}{2(d - a)^2} \sigma_b^2 \right) \\ &= q + a + \lambda + \left(\log \left(\frac{d - a}{\lambda} \right) - \frac{1}{2(d - a)^2} \sigma_b^2 \right) G, \end{aligned}$$

and taking the outer variance ($V[G] = \lambda$ for a $\text{Poisson}(\lambda)$ r.v.)

$$V_G \left[E_{u|G} \left[\hat{L}_{\bar{m}}^{\text{Pois}}(\theta) \middle| G \right] \right] \approx \lambda \left(\log \left(\frac{d - a}{\lambda} \right) - \frac{\sigma_b^2}{2(d - a)^2} \right)^2.$$

The result follows. □